**Arrays**

**let/const <arrayName> = [value1, value2, value3,….];**

- Arrays in JavaScript are also mutable (and also alias). **Note that even with const arrays, we can still change its elements. We just can’t assign new values (or new arrays) to that element**.

- Like lists in Python, changing arrays in functions also change the original array.

- We can directly pass the array to functions with its name.

1. **Indexes**

Indexes are similar to Python also. If we try to access the indexes outside the scope of the list, we will get **undefined**.

However. if we assign these to some values, JS will automatically extend the array to add the value to that index.

1. **Array Properties**

.length

1. **Array Methods**

**Note:** Some methods changes the array, but do not return anything. If we assign the calling of these methods (functions), we will get undefined, which is a default return of functions.

.push(ele1, ele2, …) 🡪 Add items to the end of array

.pop() 🡪 Like Python. Takes no argument, **remove & return last element**

.join()

.slice(start = head, end = remain) 🡪 Returns a portion copy, a new array, [start,end). Copy all .slice()

.splice()

.shift() 🡪 Takes no argument, **remove & return first element**

.unshift(elenew) 🡪 **Add first element** to array

.concat()

.indexOf(element) 🡪 Return index of element. -1 if none.